

Claims

What is claimed is:

1. A method for tracing an instrumented program, comprising:
 - triggering a probe in the instrumented program;
 - obtaining an original instruction associated with the probe;
 - loading the original instruction into a scratch space; and
 - executing the original instruction in the scratch space using the thread, wherein executing the original instruction results in placing the instrumented program in a state equivalent to natively executing the original instruction.
2. The method of claim 1, further comprising:
 - emulating a location dependent instruction to determine a value of a program counter if the original instruction is a control-flow instruction wherein semantics of the location dependent instruction depend on a location of the original instruction within the instrumented program..
3. The method of claim 1, further comprising:
 - incrementing a value of a program counter using a size of the original instruction.
4. The method of claim 1, further comprising:
 - determining a value of a program counter; and
 - loading the value of the program counter into the scratch space.
5. The method of claim 4, wherein determining the value of the program counter comprises:
 - incrementing the value of the program counter using a size of the original instruction if the original instruction is not a control-flow instruction; and
 - emulating a location dependent instruction to determine the value of the program counter if the original instruction is a control-flow instruction.

6. The method of claim 1, furthering comprising:
loading a control transfer instruction into the scratch space prior to executing the original instruction.
7. The method of claim 6, wherein the control transfer instruction includes a value of a program counter.
8. The method of claim 1, wherein the probe corresponds to a trap.
9. The method of claim 1, wherein obtaining the original instruction comprises:
searching a look-up table using a program counter value, wherein the look-up table comprises the original instruction associated with the probe and an address associated with the original instruction.
10. The method of claim 1, wherein the scratch space is allocated on a per-thread basis.
11. The method of claim 1, wherein the instrumented program is executed on multi-thread architecture.
12. The method of claim 1, wherein executing the original instruction comprises single-stepping the original instruction.
13. A system for tracing an instrumented program, comprising:
a thread configured to execute the instrumented program;
a look-up table arranged to store an address and a corresponding original instruction;
a trap handler configured to halt execution of the thread when a trap instruction is encountered and using an address of the trap instruction to obtain the corresponding original instruction from the look-up table;
a scratch space arranged to store the original instruction; and
an execution facility for executing the original instruction to obtain data.

14. The system of claim 13, further comprising:
a buffer for storing the data.
15. The system of claim 13, further comprising:
a tracing framework emulating a location dependent instruction to determine a value of a program counter if the original instruction is a control-flow instruction.
16. The system to claim 13, wherein the trap handler sets the value of the program counter to the value of a next address immediately the address of the trap instruction after executing the original instruction.
17. The system of claim 13, wherein the trap handler increments a value of a program counter using a size of the original instruction if the original instruction is not a control-flow instruction.
18. The system of claim 13, wherein the scratch space is allocated on a per-thread basis.
19. The system of claim 13, wherein the instrumented program is executed on multi-thread architecture.
20. The method of claim 1, wherein executing the original instruction comprises single-stepping the original instruction.